# Research report 149

## MULTIRESOLUTION IMAGE SEGMENTATION

**Abhir Bhalerao**
**Roland Wilson**

(RR149)

The segmentation of images into regions that have some common property is a fundamental problem in image processing. Traditional approaches to segmentation have been either edge based or region based. It is clear, however, that a multiresolution approach, where image properties are measured at varying scales, is the only way to overcome the inherent uncertainty between the class of a region and its position. A review is presented of both boundary and region based schemes concentrating on multiresolution methods. The results of an investigation into a set of rotation-invariant operators are presented. Taking boundary information as input, these operators can detect features such as corners, and could be employed to provide shape information in a segmentation scheme. A new quad-tree segmentation method is described and some initial results presented.

This work has been supported by SERC (UK) and Shell UK.

Department of Computer Science
University of Warwick
Coventry CV4 7AL
United Kingdom

September 1989

## Table of Contents


Introduction ..... 1

1. Segmentation Approaches ..... 2

    1.1 Boundary based segmentation ..... 2

    1.2 Region based segmentation ..... 5

    1.3 Multiresolution in segmentation ..... 6

    1.4 Texture segmentation ..... 8

2. Boundary, Curvature and Corners ..... 9

    2.1 Orientation Estimation and Normalisation ..... 9

    2.2 Orientation Inconsistency ..... 13

    2.3 Rotation-Invariant Operators ..... 14

3. A New Quad-tree based segmentation algorithm ..... 26

4. Conclusions and Proposals for future work ..... 28

References ..... 29

Figures ..... 32

Photographs ..... 36

# Multiresolution Image Segmentation

*Abhir Bhalerao,*
*Roland Wilson.*

Department of Computer Science.
The University of Warwick,
Coventry,
England.

## Introduction

Many natural images consist of regions that have some common property. This may be a constant or smoothly varying grey level or some textural quality. A fundamental problem in image processing is to segment such images into these regions.

There is inherent uncertainty in the task of segmentation: the spatial smoothing process required to determine a representative class for a region inevitably makes the location of the boundary of that region uncertain. The more smoothing that is applied to gain a better estimate of class the less certain the location of the boundary becomes.

The different segmentation methods that have been proposed have all been striving to overcome this uncertainty, whether knowingly or not. The many schemes that have been proposed for segmentation may be grouped into two main areas: boundary based and region based. It is clear, however, only multiresolution approaches to segmentation, where image properties are measured at varying scales, can overcome problems of uncertainty [1].

Boundaries and regions are complementary and of particular interest is to combine a boundary based approach with a region based approach. With this in mind Section 2 of this report describes an investigation into boundary and curvature representation and description. Particular interest was paid to a set of rotation invariant operators for feature extraction as described by Lenz [2] and Danielsson and Ye [3]. These operators are defined as complex convolutions and take a boundary estimate of an image that includes orientation information [4] as input. These operators can detect discontinuities in the

boundary information such as corners thus providing a higher level description of features within the image.

This report begins with a review of some notable approaches to image segmentation, concentrating on grey level segmentation and multiresolution methods. Section 2, as described above, looks at boundary and curvature. In Section 3 a new quad-tree based segmentation is described. The report concludes with proposals for future work.

## 1. Segmentation Approaches

Thresholding is the simplest way to segment an image: all pixels within a certain range of values are taken to belong to the same class. This method works well for simple tasks such as the identification of industrial 'widgets' from uniform backgrounds for which a lot of the early segmentation work was aimed.

Since this early work there have been many methods suggested to tackle segmentation of more general images where assumptions about smooth intensity variation and sharp, well defined boundaries do not apply, and contain texture and noise. There are two main groupings of these methods: boundary based and region based, with multiple resolutions being employed in both areas.

### 1.1. Boundary based segmentation

Boundary information (edges and lines) has been shown to play an important part in the low level processing of the human visual system [5] and thus ultimately in the segmentation of objects from complicated scenes [6]. This link between object boundaries and their interpretation is the motivation behind the development of edge detection methods [7] and their use in segmentation algorithms.

The best edge detection methods, however, can only result in isolated, local edge elements. Thus there is a second problem of grouping such elements together into complete boundaries that represent objects in the image.

#### 1.1.1. Edge detection

All edge detection methods convolve the image with spatial kernels to give a maximum response where there are gradients of intensity in the grey level image. There are many methods that use 3 by 3 spatial masks: Sobel, Kirsch, Frei-Chen etc. (see Gonzalez and Wintz [8] for a review).

With all these methods the aim is to gain locality in the frequency domain so as not to detect noise and fine textures, while maintaining good spatial localisation. Here again there is uncertainty, since by the uncertainty principle of signal analysis [9], to gain

better noise immunity requires greater spatial averaging and hence reduced spatial localisation.

The use of filters covering a range of scales, from the original work of Rosenfeld and Thurston [10] and later by Marr [7] to obtain spatial consistency was a notable development in edge detection methods. The rigorous mathematical approach of Canny [11] was important in setting new standards of edge detection.

The work of Hubel and Wiesel [5] also showed the existence of mechanisms in the primary visual cortex which are sensitive to *orientations* of edges and lines. Recognition of this lead to the work of Granlund and later Knutsson et al [12] and [4] who developed a quadrature filter based, orientation estimation method which provides a combined edge-line detection and orientation estimate. This technique is used in the rotation invariant operation work described in the next section.

### 1.1.2. Boundary linking

A variety of methods for linking up of the isolated elements that result from an edge detection scheme have been suggested. These methods vary in the amount of a priori knowledge that is required, although most require that the edge detection result be thresholded to produce a binary image.

Line linking involves a heuristic search where each edge element is linked with a near neighbour according to some similarity criterion such as edge strength or orientation.

Graph theoretic methods view the problem of completing boundaries as a graph searching problem where the required boundary is a least-cost path between nodes of a weighted graph. The nodes of the graph are the isolated edge elements and the weights of arcs set to be the measurement of similarity. Ballard and Brown [13] provide a good review of such methods.

The Hough Transform [Hough 1962] is a well documented technique in image processing. The transform is applied on an edge map and can be used for lines, circles and

curve detection given a parametric description of the type of curve to be detected.

For example, it is possible to find straight lines in the image plane given only isolated points or edge elements. In the image plane $(x,y)$ lines may be expressed as $y = ax + b$, giving an infinite number of possible lines passing through any point $(x',y')$. Transforming to parameter space $(a,b)$ by the equation $b = -xa + y$ now says that all points that lie on a particular line $y = a'x + b'$ will be represented by the point $(a',b')$ in the parameter space. Dividing the $(a,b)$ plane into a discrete number of accumulator cells and then incrementing a and solving for b using $b = -xa + y$ for each $(x',y')$ of the edge image, results in maxima in the $(a,b)$ plane which correspond to the points that are collinear in the image. The Hough transform can been generalised to other curves and closed shapes [14].

All these methods have the limitation that they are sensitive to noise and cannot cope with the presence of texture.

### 1.2. Region based segmentation

All region based methods attempt to find sets of pixels that have some common property and thus correspond to the objects in the image. The idea of similarity of image property leads to the use of statistical methods to determine the class of a pixel.

A histogram of an image property such as intensity will reveal peaks that correspond to the representative values of the component regions in the image. Statistical classification involves estimating parameters of the component region populations such as mean and variance, and then determining optimal thresholds to separate out these regions.

Heuristically derived clustering methods may also be used to automatically determine class values from global statistics. A clustering technique was successfully used by Spann and Wilson [15] as part of their quad-tree segmentation algorithm.

The main problem with statistical classification procedures is that the histogram does not preserve spatial information, it is a global statistic. It is essential therefore that spatial information is incorporated into a classification scheme, as in split-and-merge

methods.

Split and Merge methods, first proposed by Pavlidis and used in the work with Chen [16], use statistics to determine the uniformity of a region. This region is then split into 4 if not uniform and the test reapplied to each sub-region and the process repeated. The split regions are then merged according to adjacency rules and further statistical measures. This algorithm is notable because it uses a pyramidal data structure and its determination of a 'boundary ambiguity zone', also features of the work of Spann and Wilson.

Traditional region growing methods attempt to grow regions from selected starting points with some form of similarity criterion being used for adjacent pixels. The statistics of the region being formed is updated with the addition of each new pixel. This process is further continued until all pixels are part of some region. The process is continued by merging neighbouring regions to form larger regions, based on similarity criteria.

Region growing methods have combined spatial information (pixel adjacency) with statistical measures of similarity. The drawbacks are that the starting points have to be input, the processing is sequential in operation, and that it cannot be guaranteed that given different starting points the methods will result in the same segmentation.

### 1.3. Multiresolution in segmentation

Multiple scales in segmentation allow local and global information to be combined. Multiresolution methods often use pyramidal data structures such as quad-trees. A quad-tree of an image $d(i,j)$, $0 \le i,j \le N$ of size $(N \times N)$, $N = 2^m$ denoted by $q(i,j,k)$, $0 \le k \le m$, $0 \le i,j \le 2^{m-k}$, can be defined iteratively as:

$$q(i,j,k) = 0.25[\, q(2i,2j,k-1) + q(2i+1,2j,k-1) + q(2i,2j+1,k-1) + q(2i+1,2j+1,k-1)\,], \; k>0$$

$$q(i,j,0) = d(i,j,0) \qquad (1.1)$$

Note that the quad-tree is a non-overlapping, 2x2 averaging process where the top-node of the tree represents the mean value of the image. Quad-trees are not restricted to grey level data. The quad-tree offers a fast smoothing operation with each successive

level trading off the reduction in variance of each homogeneous region against spatial resolution. At some level $m' < m$, however, it is possible to achieve maximum smoothing while maintaining sufficient resolution to ensure accurate segmentation [15]. The quad-tree data structure is illustrated in figure 1.1.

Pavlidis and Chen employed a linked pyramid data structure in their split-and-merge algorithm [16]. Such a structure uses 4x4 averaging, again producing $m$ levels for an image of size $N = 2^m$. The 4x4 areas overlap with each child having 4 parents and each parent 16 children. There are two types of association that can be made within a pyramid: between neighbouring nodes on each level and between parents and children. Pavlidis and Chen determine parent/child links resulting in regions in the image being represented by a root node at some level $m'$, $m' < m$.

The linked pyramid algorithm of Burt et al [17] builds a pyramid setting links between children and parents according to the minimum absolute distance. The values in the pyramid are then repeatedly updated, bottom-up, replacing each parent by a weighted average, based on the population that each node represents, using the links made in the previous step. This achieves local spatial smoothing. Next the segmented class values are passed top-down which shifts region boundaries in a direction that make the contents more homogeneous.

Hong and Rosenfeld [18] extended this approach by attaching weights to child/parent links. The idea behind this is to defer a classification decision until more information is available. In both methods the number of regions have to be known a priori.

The quad-tree segmentation method of Spann and Wilson [15], shown in figure 1.2, does not require prior knowledge about the number of regions. Having built the quad-tree it then uses a clustering method on the histogram of some level $m' < m$, where $m'$ is determined by the size of regions in the image. The clustering method works iteratively by moving probability masses to their centre of gravity within a sliding window. The selected level $m'$ is then classified from the classes output by the clustering method. At this point assumptions about spatial coherence and region homogeneity across scales of the quad-tree, allow the classification at this level to be valid for lower levels. Having

classified the level below, a boundary region is identified. This region of uncertainty is all those nodes that have an eight-neighbour not of the same class as itself. The boundary region is then dilated and then smoothed with a linear filter whose spatial width is determined an estimate of intra-regional signal-to-noise, on the current level. After smoothing the boundary nodes are classified using a nearest class mean criterion. Isolated nodes are then removed and the process repeated for the next level until full spatial resolution is restored.

This method is clearly superior to previous statistical approaches and good results are obtained even at low signal-to-noise ratios. Uncertainty is overcome by making assumptions about class membership across scales. The drawbacks are the selection of the optimal level $m'$ to begin the segmentation and the isotropic nature of the smoothing process which tends to erode regions that have promontories.

The recent work of Horne and Spann [19] combines the linked pyramid approaches of Hong, Rosenfeld, Burt etc., and the boundary refinement step of Spann and Wilson. It again can work on images that have an arbitrary number of regions and improves on the lower limit of region size, giving good results for poor signal-to-noise ratios. It does, however, make assumptions about the convexity and aspect ratios of regions.

## 1.4. Texture segmentation

Nothing so far has been said about the segmentation of texture, where the image intensity alone at one point cannot be used as an estimation of region property. One approach is to characterise each pixel by the textural properties of its neighbourhood, yielding a description consisting of a vector quantity. For example local properties of spatial frequency and orientation may be used.

The concept of edge detection is still valid in a multi-dimensional space where edges will exist in more than one attribute. Statistical classification, clustering and region similarity measures also can be extended to multi-dimensional data.

Spann and Wilson have successfully extended their grey level segmentation algorithm to texture by multi-dimensional data derived from optimal spatial frequency feature

sets [20].

## 2. Boundary, Curvature and Corners

The work described in this section was motivated by two things: first the recognition of the importance of boundary information and higher level features in an image, and second the need to incorporate some form of structural information into a segmentation algorithm.

### Curvature

Curvature has received a lot of interest in image processing. Attneave [21] demonstrated that a silhouette image is still recognizable because of curvature changes along its boundary. Consequently the detection of points of high curvature is often used in computer vision as, for example, in point selection in polygon approximation and curve fitting, pattern matching, or determination of 3-d shape fro 2-d line information.

Curvature approaches often take for granted that line/edge detection has been performed, sometimes incorporating orientation information, as in [22]. This usually assumes that the edge detection has been thresholded restricting its use to simple object-background tasks, as problems such as noise are ignored. The alternative approach, as used by Bårman [23], is to estimate curvature without thresholding. Bårman uses an orientation estimate (Knutsson et al [12]) as an input image and performs further filtering to determine rate of angular change in this image.

The work described here is not directly concerned with curvature estimation, although points of high curvature, such as corners, can be identified. It has similarities with the work of Bårman it that it uses the same orientation estimate as the starting boundary representation.

### 2.1. Orientation Estimation and Normalisation

The orientation estimation method is taken from Knutsson et al [12]. A brief description follows.

An important property of the Fourier Transform is that a roughly 1-dimensional neighbourhood, orientated at an angle $\theta$ will result in energy being concentrated around a line through the origin orthogonal to $\theta$ in the spatial frequency domain, see figure 2.1. The orientation estimation procedure uses a set of filters (at least 3) to collect energy from partitions of the Fourier domain.

If the image $f(x,y)$ is real then its transform $F(u,v)$ will be Hermitian symmetric i.e. the real part of $F$ is an even function and the imaginary part an odd function. The orientation filters $H_k(u,v)$ spatially represent a combined line (even) and edge (odd) filters:

$$H_k(u,v) = H_{ke}(u,v) + H_{ko}(u,v) \tag{2.1}$$

$$H_{ke}(u,v) = H_{ke}(-u,-v) \tag{2.2}$$

$$H_{ko}(u,v) = sgn\,[\cos(\alpha-\phi_k)]H_{ke}(u,v) \tag{2.3}$$

where $\alpha = arg(u + jv)$ and $\phi_k$ is the direction of the filter $k$, $0 \le k < K$. Notice that the filters $H_k$ are in *quadrature* i.e. that they take up energy in half of the Fourier domain only, figure 2.2. $H_{ke}$ can be chosen as:

$$H_{ke}(u,v) = W(\rho)\cos^2(\alpha-\phi_k),\ \rho = \sqrt{u^2+v^2} \tag{2.4}$$

$W(\rho)$ is a radial weighting function determining the frequency selectivity of the orientation estimate. The filters are set evenly over half the spatial frequency plane. The orientation estimate is given by the sum of the magnitudes of the filter responses:

$$o(x,y) = \sum_{k=0}^{K-1} |g_k(x,y)|\, e^{j2\phi_k} = re^{j\psi} \tag{2.5}$$

The doubling of $\theta_k$ is deliberate producing at each point the estimate $\psi$ which is twice the local angle and $r$ is a certainty measure of how reliable the estimate is. The double angle is essential to make operations such as averaging meaningful.

Photo 3 shows the magnitude of an orientation estimate of the *girl* image (photo 1). The result demonstrates the combined line/edge detection of the estimate.

### 2.1.1. Fast orientation estimate

Although the Knutsson orientation estimation is accurate it is computationally expensive and therefore slow. Using a set of 4 selection filters provides an estimate which is accurate to within a few degrees, but requires 143 operations per image pixel (Todd [24] ).

Todd [24] has developed a fast orientation method which only requires 3 multiplications per pixel which is a theoretical increase in speed of 48 times. This estimator models intensity over a block of 4 pixels by bilinear interpolation. The estimated orientation is given by minimising the integral of the square of the gradient in a given direction.

Given the 4 pixels in a block: $a, b, c, d$ the estimate of orientation $\psi$ can be expressed as a vector:

$$\underline{v}_\psi = \begin{bmatrix} 2(b-c)(d-a) \\ (b-c)^2-(d-a)^2 \end{bmatrix} \tag{2.6}$$

The argument of $\underline{v}_\psi$ is the double angle $\psi$ and its magnitude the certainty measure $r$

$$\psi = arg\left[\underline{v}_\psi\right] \tag{2.7}$$

$$r = (b-c)^2+(d-a)^2 \tag{2.8}$$

The fast estimator is less accurate and more sensitive to noise because of its simple 2x2 nature. Before running the estimator most images have to be first low pass filtered to smooth out sharp edges and noise.

Photo 4 shows the magnitude of the fast orientation estimate on the *leaf* image shown in photo 2. In comparison to the slower orientation estimation the response is more high pass. It is also essentially an edge detector.

### 2.1.2. Quad-tree Normalisation

A quad-tree based normalisation method was used on the orientation estimate to increase the magnitude of the estimate where it was low in the image. This method is based on the one used by Calway [25]. It works by determining a scaling factor for a

local region based on the values within that region. The hierarchical nature takes into consideration values of the region extending out from the locality. The method is as follows:

(1) A quad-tree $q(i,j,k)$ of the scaled magnitude of the orientation estimate $o'(i,j)$, $0 \le i,j \le N, N = 2^m$ is built

(2) Working from the top, each level a simple recursion on the scale index is performed [26]:

$$q'(i,j,m) = q(i,j,m) \tag{2.9}$$

$$q'(i,j,k-1) = \beta q(i,j,k-1) + (1-\beta) q(i/2,j/2,k),\ 0 \le i,j \le 2^{m-k},\ k < m \tag{2.10}$$

(3) Finally the normalisation of the original is performed by:

$$n(i,j) = \frac{o'(i,j)}{q'(i,j,0)} \tag{2.11}$$

The value of $\beta$, $0 \le \beta \le 1.0$ determines the locality of the normalisation. Small values of $\beta$ increase the effective area and large values reduce this area.

The problem with using the quad-tree is that down propagation introduces blocking artifacts in the result. To overcome this the method is modified to use intermediate nodes, an idea taken from Clippingdale and Wilson [27] who were faced with the same problem. These intermediate nodes do not themselves form a tree but straddle the boundaries of blocks which are the artifacts of the original tree, see figure 2.3.

On level $k$ of the tree there are $(2^{m-k})^2$ original nodes. To this $(2^{m-k} - 1)^2$ intermediate nodes are introduced that each lie centrally between four original nodes. The children of an intermediate node are taken to be these four closest to it from the original level.

$$i(i,j,k) = 0.25[\, q(i,j,k) + q(i+1,j,k) + q(i,j+1,k) + q(i+1,j+1,k)\,],\ 0 \le i,j \le 2^{m-k} - 1 \tag{2.12}$$

The above procedure is modified as follows:

(1) After the quad tree has been built the intermediate nodes are derived for each level

(2) After the first propagation down from level $m$ to $m-1$, new intermediate nodes $i'(i,j,k)$ are calculated

(3) The original intermediate nodes are then updated as follows:

$$i(i,j,k) = \beta i'(i,j,k) + (1-\beta) i(i,j,k) \tag{2.13}$$

(4) Then the current level is modified further by redistributing the intermediate node values to its children:

$$q'(i,j,k) = 0.5q'(i,j,k) + 0.5i'(i,j,k) \tag{2.14}$$

Photo 5 shows the quad-tree normalised response of that shown in photo 4. Notice how the low magnitudes are brought up revealing the fine, fractal like structure of the leaf.

## 2.2. Orientation Inconsistency

Having obtained an orientation estimate, as described above, the next step was to determine ways to identify the features of interest, such as corners and other discontinuities in the boundary.

Todd [24] has used a measure of local consistency of orientation, see also [12]. This is determined for a square neighbourhood $(n \times n)$ of the orientation estimate $o(i,j)$. Two measures are calculated in the area: the magnitude of the average orientation and the average of the magnitudes.

$$\hat{o} = \frac{1}{n^2} \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} o(i,j) \tag{2.15}$$

$$e = \frac{1}{n^2} \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} |o(i,j)| \tag{2.16}$$

The ratio of the magnitude of the average orientation and the average energy gives a coefficient of consistency $c$, $0 \le c \le 1.0$.

$$c = \frac{|\hat{o}|}{e} \tag{2.17}$$

Note that the double angle representation of $o(i,j)$ makes the average $\hat{o}$ a sensible result. This consistency measure can obviously be used to determine inconsistency by using $(1.0 - c)$. If this is weighted by the local energy $e$ a better measure is obtained:

$$i = \left[1.0 - \frac{|\hat{o}|}{e}\right] e = e - |\hat{o}| \tag{2.18}$$

The values of inconsistency can be calculated for each point in the orientation image using a circular window of size $r$. The whole operation is equivalent taking the difference between the low pass filtering the magnitude of the orientation estimate, and the magnitude of the low pass filtering of the orientation estimate. If $L\{\}$ is the low pass filtering operation then:

$$i(i,j) = L\{|o(i,j)|\} - |L\{o(i,j)\}| \tag{2.19}$$

Photo 6 shows an inconsistency estimate of the orientation map of the *girl* image obtained using equation (2.19).

## 2.3. Rotation-Invariant Operators

The ability of the human visual system to recognise patterns independent of position (shift), size (scale) and orientation (rotation) has been of interest in computer vision and pattern recognition for many years.

This work looks at a particular set of operators which can detect patterns formed by boundaries, such as junctions and corners, invariant of their orientation. These operators are conceptually simple and can be implemented by 2-D filtering.

Lenz [2] has already shown how such operators can be used for the detection of patterns in a neighbourhood. Examples of their application are given in the work of [28] and Danielsson and Ye [3]. In the work presented here is these operators have been applied

to the vector orientation estimate rather than the grey level image itself.

**Definition**

An nth order rotational operator is a complex function with a circular symmetric magnitude envelope and a phase that changes $n$ times round the circle, see figure 2.5. A rotation-invariant operator of order $n$ is defined as:

$$h_n(\rho,\theta) = h(\rho)e^{jn\theta} \tag{2.20}$$

where $(\rho,\theta)$ are polar coordinates and $h(\rho)$ is a radial envelope function. If now the operator is run on the complex orientation image:

$$o(\rho,\theta) = r(\rho,\theta)e^{j\psi(\rho,\theta)} \tag{2.21}$$

the output at each point is given by the inner product between the image and the operator coefficients:

$$\begin{aligned} g_{n1} &= \sum_{\rho}\sum_{\theta} o(\rho,\theta)\cdot h_n(\rho,\theta) \\ &= \sum_{\rho}\sum_{\theta} r(\rho,\theta)h(\rho)\cos(\psi(\rho,\theta) - n\theta) \\ &= \sum_{\rho}\sum_{\theta} Re\{o(\rho,\theta)h_n{}^*(\rho,\theta)\} \end{aligned} \tag{2.22}$$

A second operator $h'_n$ which is rotated $\pi/2$ to the first is used:

$$h'_n(\rho,\theta) = h_n(\rho,\theta + \pi/2) \tag{2.23}$$

The second output is given by:

$$\begin{aligned} g_{n2} &= \sum_{\rho}\sum_{\theta} o(\rho,\theta)\cdot h_n(\rho,\theta + \pi/2) \\ &= \sum_{\rho}\sum_{\theta} r(\rho,\theta)h(\rho)\cos(\psi(\rho,\theta) - n\theta + \pi/2) \\ &= \sum_{\rho}\sum_{\theta} -r(\rho,\theta)h(\rho)\sin(\psi(\rho,\theta) - n\theta) \\ &= \sum_{\rho}\sum_{\theta} Im\{o(\rho,\theta)h_n^*(\rho,\theta)\} \end{aligned} \tag{2.24}$$

Combining the two outputs $g_{n1}$ and $g_{n2}$ into a single complex value gives:

$$g_n = g_{n1} + jg_{n2}$$

$$= o(\rho,\theta)h_n^*(\rho,\theta) \tag{2.25}$$

**Rotational invariance**

What happens if the input image is rotated by an angle $\phi$, giving the new image $o'$ ?

$$o'(\rho,\theta) = o(\rho,\theta+\phi) \tag{2.26}$$

Applying the operator, from equations (2.22)-(2.24) we have:

$$\begin{aligned} g'_n &= \sum_\rho \sum_\theta o'(\rho,\theta)h_n^*(\rho,\theta) \\ &= \sum_\rho \sum_\theta o'(\rho,\theta)h(\rho)\Big[\cos n\theta - j\sin n\theta\Big] \end{aligned} \tag{2.27}$$

replacing $\theta$ by $\theta' = \theta + \phi$ yields:

$$\begin{aligned} g'_n &= \sum_\rho h(\rho)\sum_{\theta'} o'(\rho,\theta')\Big[\cos n(\theta'-\phi) - j\sin n(\theta'-\phi)\Big] \\ &= \sum_\rho h(\rho)\sum_{\theta'} o(\rho,\theta')h_n^*(\rho,\theta'-\phi) \\ &= e^{-jn\phi}g_n \end{aligned} \tag{2.28}$$

Hence the result reflects the rotation by a similar phase change in the output.

**A suitable envelope function**

After some experimentation an annulus shaped envelope function $h(\rho)$ was chosen.

$$h(\rho) = \cos^2\left[\pi/2\ \frac{\ln(\rho/\rho_0)}{\ln(\rho_1/\rho_0)} - \pi/2\right] \tag{2.29}$$

where $\rho_0$ is the inner radius and $\rho_1$ radius at the the peak value. The profile is illustrated in figure 2.4.

**Auto-Normalisation**

One of the problems with $h_n$ is that it responds to linear features, producing a 'tram' lines either side of a boundary line. The response of $n = 1$ for example is zero on the boundary and lines either side with opposite phases. This unwanted response is removed

by the use of a non-linear normalisation process suggested by Knutsson [private correspondence]:

$$g_n = (h_n * o)(|h_n| * |o|) - (|h_n| * o)(h_n * |o|) \tag{2.30}$$

where * means convolution, and $|h_n|$ and $|o|$ are magnitudes of the spatial complex functions. It is easy to see that if the filter had no angular variation or if the orientation field was constant then the result would be zero. The result for other cases is not so obvious. One way to look at it is to take the four terms separately: $(h_n * o)$ is the actual filter response that produces the undesired tram line effect. $(|h_n| * |o|)$ is in effect a band pass filtering of the magnitude of the orientation estimate which will spread out the boundaries in the image. $(|h_n| * o)$ is the low pass of the orientation estimate. $(h_n * |o|)$ will produce a tram line effect similar to the first term except that the phases will be reversed. The multiplications and differencing should then produce a zero response on straight lines in the orientation estimate.

Photo 7 shows the result of applying operator $n = 1$ to the orientation estimate of the *leaf* image. It shows clearly the response to corner like features and the undesirable 'tram' lines response. Photos 8 and 9 show auto-normalised responses to operators $n = 1$ and $n = 2$ respectively. The high responses in photo 9 correspond to roughly circular features.

### 2.3.1. Frequency characteristics

The Fourier transform of the operator functions $h_n$ shows that the angular variation is carried through to the frequency domain.

$$H(u,v) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h_n(x,y)e^{-\pi j(ux+vy)}\,dxdy \tag{2.31}$$

In polar coordinates $x = \rho\cos\theta$, $y = \rho\sin\theta$ and $u = w\cos\phi$, $v = w\sin\phi$. Also $dxdy = \rho\, d\rho d\theta$:

$$H(u,v) = \int_0^{\infty}\rho\, h(\rho)\int_{-\pi}^{\pi} e^{-jn\theta}e^{j(u\rho\cos\theta + v\rho\sin\theta)}\,d\theta d\rho$$

$$H(u,v) = \int_0^{\infty}\rho\, h(\rho)\int_{-\pi}^{\pi} e^{-jn\theta}e^{jw\rho\cos(\theta-\phi)}\,d\theta d\rho \tag{2.32}$$

Now replacing $\theta$ by $\theta + \phi$:

$$= \int_0^{\infty} \rho\, h(\rho) \int_{-\pi}^{\pi} e^{-jn(\theta+\phi)} e^{jw\rho\cos(\theta)}\, d\theta d\rho$$

$$= e^{-jn\phi} \int_0^{\infty} \rho\, h(\rho) \int_{-\pi}^{\pi} e^{-jn\theta} e^{jw\rho\cos\theta}\, d\theta d\rho \qquad (2.33)$$

The last integral is in fact the nth order Bessel function $J_n(z)$ [ 29] from:

$$J_n(z) = \frac{1}{\pi}\int_0^{\pi} \cos(z\sin\theta - n\theta)\, d\theta$$

$$= \frac{j^{-n}}{\pi}\int_0^{\pi} e^{jz\cos\theta}\cos(n\theta)\, d\theta$$

$$= \frac{j^{-n}}{2\pi}\int_{-\pi}^{\pi} e^{jz\cos\theta}\, e^{-jn\theta}\, d\theta \qquad (2.34)$$

The proof of the change in integration limits from $[0,\pi]$ to $[-\pi,\pi]$ and the replacement of $\cos n\theta$ by $e^{-jn\theta}$ in equation (2.34) is as follows:

$$J_n(z) = \frac{j^{-n}}{2\pi}\int_{-\pi}^{\pi} e^{jz\cos\theta}\, e^{-jn\theta}\, d\theta$$

$$= \frac{j^{-n}}{2\pi}\left[\int_{-\pi}^{0} e^{jz\cos\theta}\, e^{-jn\theta}\, d\theta + \int_0^{\pi} e^{jz\cos\theta}\, e^{-jn\theta}\, d\theta\right] \qquad (2.35)$$

replacing $e^{-jn\theta}$ by $(\cos n\theta - j\sin n\theta)$ gives:

$$J_n(z) = \frac{j^{-n}}{2\pi}\left[\int_{-\pi}^{0} e^{jz\cos\theta}\cos n\theta\, d\theta - \int_{-\pi}^{0} e^{jz\cos\theta}\, j\sin n\theta\, d\theta\right.$$

$$\left. + \int_0^{\pi} e^{jz\cos\theta}\cos n\theta\, d\theta - \int_0^{\pi} e^{jz\cos\theta}\, j\sin n\theta\, d\theta\right]$$

Now since cos() is an even function ($E$), and sin() an odd function ($O$) and $e^{\cos\theta}$ an even function of $\theta$:

$$J_n(z) = \frac{j^{-n}}{2\pi}\left[\int_{-\pi}^{0} E\times E\, d\theta - \int_{-\pi}^{0} E\times O\, d\theta + \int_0^{\pi} E\times E\, d\theta - \int_0^{\pi} E\times O\, d\theta\right] \qquad (2.36)$$

where the second and fourth integrals cancel out leaving two integrals having the same value:

$$J_n(z) = \frac{j^{-n}}{2\pi} \cdot 2 \int_0^{\pi} e^{jz\cos n\theta} e^{-jn\theta}\, d\theta \tag{2.37}$$

Equation (2.33) now simplifies to:

$$H(u,v) = \left[ \frac{e^{-jn\pi/2}}{2\pi} \right] e^{-jn\phi} \int_0^{\infty} \rho h(\rho) J_n(w\rho) d\rho \tag{2.38}$$

The last integral being the nth order Hankel transform of the radial variation $h(\rho)$ where $J_n$ is the nth order Bessel function. The importance of equation (2.38) is that it reveals that the Fourier transform of the rotational invariant operators maintain the same phase variation [3]. Note that there is an offset of $n\pi/2$ in the phase.

What this implies is that it is possible to specify the operator characteristics either spatially or in the frequency domain. Both methods were adopted when looking at the results of several order operators together to either make their spatial extent the same of their frequency selectivity the same.

### 2.3.2. Pattern detection

It is easy to interpret the results of the operators of order 1 and 2 as detecting corner like patterns and circles respectively. The interpretation of the higher order operators is not so obvious. The results around junction points which have more than 2 arcs, like "Y" shapes commonly found in natural images, is uncertain. This leads to the question of what information can be gained by looking at the results of several of the orders together and also what phases of these coefficients tell us.

#### Absolute orientation

If a junction point in the orientation image is modelled as radial pattern as a set of $M$ orientated lines:

$$f(\theta) = \sum_{k=0}^{M-1} \delta(\theta - \theta_k) e^{j2\theta_k} \tag{2.39}$$

Note the doubling of the exponent in equation (2.39) angle. The result of applying an nth

order filter to this pattern is:

$$g_n = \sum_{k=0}^{M-1} \int_0^{2\pi} \delta(\theta - \theta_k) e^{j2\theta_k} e^{-jn\theta}\, d\theta$$

$$= \sum_{k=0}^{M-1} e^{-j(n-2)\theta_k} \tag{2.40}$$

If this pattern is rotated by $\phi$:

$$g'_n = \sum_{k=0}^{M-1} \int_0^{2\pi} \delta(\theta - \theta_k - \phi) e^{j2(\theta_k + \phi)} e^{-jn\theta}\, d\theta$$

$$= \sum_{k=0}^{M-1} e^{-j(n-2)(\theta_k + \phi)}$$

$$= e^{-j(n-2)\phi} \sum_{k=0}^{M-1} e^{-j(n-2)(\theta_k)}$$

$$= e^{-j(n-2)\phi}\, g_n \tag{2.41}$$

Hence a rotation of $\phi$ introduces a phase of $-(n-2)\phi$ into each coefficient with the phase response of the operator $n=2$ being unaffected.

**Periodicity**

Consider a pattern which is periodic and has $M$ spars equally spaced around the circle at $\frac{2\pi}{M}$ intervals:

$$f(\theta) = \sum_{k=0}^{M-1} \delta(\theta - \frac{2\pi k}{M}) e^{j2\frac{2\pi k}{M}} \tag{2.42}$$

then:

$$g_n = \int_0^{2\pi} \delta(\theta - \frac{2\pi k}{M}) e^{j2\frac{2\pi k}{M}} e^{-jn\theta}\, d\theta$$

$$= \sum_{k=0}^{M-1} e^{j(\frac{4\pi k}{M})} e^{-jn\frac{2\pi k}{M}}$$

$$= 0, \quad n - 2 \neq aM$$

$$= M, \quad n - 2 = aM \tag{2.43}$$

So the periodicity of the pattern is reflected in the spacing of the non-zero coefficients centred on the $n=2$ operator.

**Closedness**

If the above model is changed by making the phase constant over a sector of the circle:

$$f_k(\theta) = e^{j2\frac{2\pi k}{M}}, \quad |\theta - \frac{2\pi k}{M}| \leq \pi/M$$

$$= 0, \quad else \tag{2.44}$$

Then the complete pattern is given by:

$$f(\theta) = \sum_{k=0}^{M-1} f_k(\theta) \tag{2.45}$$

and

$$g_n = \sum_{k=0}^{M-1} \int_{\pi/M(2k-1)}^{\pi/M(2k+1)} e^{j\frac{4\pi k}{M}} e^{-jn\theta}\, d\theta$$

$$= \sum_{k=0}^{M-1} e^{j\frac{4\pi k}{M}} \cdot \frac{1}{nj}\left[ e^{-jn\frac{(2k-1)}{M}\pi} - e^{-jn\frac{(2k+1)}{M}\pi} \right]$$

$$= \frac{2}{n} \sum_{k=0}^{M-1} e^{j\frac{4\pi k}{M}} e^{-j\frac{2kn\pi}{M}} \sin\frac{n\pi}{M}$$

$$= \frac{2\sin\frac{n\pi}{M}}{n} \sum_{k=0}^{M-1} e^{-j\frac{2k\pi}{M}(n-2)}$$

$$= 0, \quad n-2 \neq aM$$

$$= M\,\frac{2\sin\frac{n\pi}{M}}{n}, \; n-2 = aM \tag{2.46}$$

which is a similar result to equation (2.43). If the phase in each sector of the pattern is offset by $\phi$, opening out the feature so that it starts to become more circular:

$$f_k(\theta) = e^{j2\left[\frac{2\pi k}{M} + \phi\right]} \tag{2.47}$$

then from above:

$$g_n = 0, \quad n-2 \neq M$$

$$= M\frac{2\sin\frac{n\pi}{M}}{n} e^{-2\phi}, \quad n-2 = M \tag{2.48}$$

So the phase of the operator $n$=2 reflects the closedness of the pattern with the sign indicating right or left handedness of the pattern. Bigün [28] has used a similar filter to detect spiral symmetry patterns on partial derivative based orientation estimates.

**Results**

Figure 2.6 show a synthetic image which was used to test out the ideas described above. The image contains symmetric patterns having $M$ = 1 to $M$=6 spars, at 45 degree orientations and opening out at 22.5 degree intervals from 0 to 90. An orientation estimate was run on this image and then a set of rotation-invariant operators, from orders $n = -4$ to $n = 8$, applied. The coefficients were calculated at the centre points of each of the patterns. Table 1 show the estimated orientation, closedness and periodicity for some of the patterns. The 'closedness' is the argument of the operator $n = 2$.

Having obtained encouraging results form the above experiment a pattern matching exercise was tried. Several possible junction primitives were chosen, as shown in Figure 2.7, and rotation-invariant coefficients were calculated for each. Table 2 show the results for the limited set of primitives shown in Figure 2.7. A natural image was selected and an orientation estimate calculated. Using an inconsistency measure and peak detection a set of points of interest selected, at which the rotational-operators applied. By using complex correlation between these selected points and the set of primitives the best match was chosen, plus the estimates of orientation, closedness and periodicity calculated. Figure 2.8 show the result of labeling the junction features in this way.

While the results show some promise it does appear that in many natural images there is a rather high ambiguity in the results obtained with these methods. This seems to be due to: lack of orientation selectivity of the operators and variation in magnitude and curvature of the orientation estimate.

Table 1

| | Orient | Closedness | Period |
|---|---|---|---|
| 0 | 341 | 0 | 1 |
| 1 | 109 | 1 | 1 |
| 2 | 71 | 0 | 1 |
| 3 | 19 | 1 | 1 |
| 4 | 341 | 0 | 1 |
| 5 | 289 | 1 | 1 |
| 6 | 251 | 0 | 1 |
| 7 | 19 | 1 | 1 |
| 8 | 351 | 20 | 1 |
| 16 | 323 | 45 | 1 |
| 24 | 326 | 63 | 1 |
| 32 | 333 | 89 | 1 |
| 40 | 49 | 0 | 2 |
| 48 | 90 | 18 | 2 |
| 56 | 56 | 42 | 2 |
| 64 | 244 | 65 | 2 |
| 72 | 270 | 89 | 2 |
| 80 | 11 | 0 | 3 |
| 88 | 232 | 19 | 3 |
| 96 | 264 | 44 | 3 |
| 104 | 54 | 69 | 3 |
| 112 | 319 | 92 | 3 |
| 120 | 296 | 1 | 4 |
| 128 | 320 | 15 | 4 |
| 136 | 6 | 38 | 4 |
| 144 | 98 | 74 | 4 |
| 152 | 266 | 90 | 4 |
| 160 | 100 | 0 | 5 |
| 161 | 351 | 0 | 5 |

Table 1

| | Orient | Closedness | Period |
|---|---|---|---|
| 162 | 10 | 0 | 5 |
| 163 | 80 | 0 | 5 |
| 164 | 281 | 0 | 5 |
| 165 | 350 | 0 | 5 |
| 166 | 11 | 0 | 5 |
| 167 | 260 | 0 | 5 |
| 168 | 3 | 12 | 5 |
| 176 | 46 | 38 | 5 |
| 184 | 75 | 64 | 5 |
| 192 | 288 | 91 | 5 |
| 200 | 256 | 0 | 6 |
| 208 | 146 | 14 | 6 |
| 216 | 107 | 33 | 6 |
| 224 | 10 | 70 | 6 |
| 232 | 64 | 91 | 0 |

Table 2

| | Orient | Closedness | Period |
|---|---|---|---|
| 0 | 45 | 0 | 1 |
| 1 | 73 | 0 | 1 |
| 2 | 85 | 0 | 1 |
| 3 | 117 | 0 | 1 |
| 4 | 86 | 1 | 1 |
| 5 | 109 | 0 | 1 |
| 6 | 92 | 0 | 1 |
| 7 | 80 | 0 | 2 |
| 8 | 41 | 1 | 1 |
| 9 | 43 | 0 | 4 |
| 10 | 30 | 0 | 3 |
| 11 | 112 | 1 | 1 |
| 12 | 106 | 1 | 1 |
| 13 | 84 | 1 | 1 |
| 14 | 60 | 2 | 1 |
| 15 | 39 | 0 | 3 |
| 16 | 44 | 0 | 5 |
| 17 | 0 | 0 | 2 |
| 18 | 16 | 1 | 2 |

## 3. A New Quad-tree segmentation algorithm

This section presents the initial steps of a new quad-tree based segmentation scheme.

### Optimal level of smoothing

At some level $m' < m$ in a quad tree of a image $N = 2^m$, it is possible to achieve maximum smoothing while maintaining sufficient resolution to ensure accurate segmentation [15]. The question arises, however, of how this level $m'$ can be determined. Assumptions can be made about region size which will give the maximum level at which regions are spatially distinct. Alternatively the segmentation process can be run several times, each time using statistics from different levels. The best result can then be selected.

One other way is to abandon the notion of the whole image being optimally represented at one level and to have areas of the image represented at different scales. In this structure large, smooth regions would be represented at high levels in the tree, while uncertain areas such as boundary regions, being represented at the lower levels.

### Heterogeneous quad-tree

When building the quad-tree if four children are part of a homogeneous region then it is to be expected that their variance would be small. If, however, they happen to straddle two different regions i.e. are part of a boundary, then their variance will be greater.

The new segmentation scheme builds a heterogeneous quad-tree by measuring the variances between child nodes. If the variance of the children is found to be greater than some factor $\alpha$ of the average variance for all groups of four children on that level then the parent node is marked. When the process is complete the lowest marked nodes are taken to be representative of the underlying region. A node on level $k$ representing a region of size $2^{m-k}$ square in the original image. By changing $\alpha$ it is possible to control the extent of the variability allowed. The procedure can be summarised as follows:

[1] Build a quad tree $q(i,j,k)$ of the image

[2] Work from level 0 to top calculating 2 by 2 block variances $v(i,j,k)$:

$$v(i,j,k) = \Big[q(2i,2j,k) - q(i,j,k+1)\Big]^2 + \Big[q(2i,2j+1,k) - q(i,j,k+1)\Big]^2 + \Big[q(2i+1,2j,k) - q(i,j,k+1)\Big]^2 + \Big[q(2i+1,2j+1,k) - q(i,j,k+1)\Big]^2 \quad (3.1)$$

[3] If $v(i,j,k) > \alpha\, \bar{v}_{k-1}$, then mark node $q(i,j,k+1)$, where $\alpha$ is a control factor and:

$$\bar{v}_k = \left[\frac{1}{2^{m-k}}\right]^2 \sum_{i}^{2^{m-k}} \sum_{j}^{2^{m-k}} v(i,j,k) \quad (3.2)$$

**Results**

Photo 10 shows the original *shapes* test image and photo 11 the image with added Gaussian white noise with mean zero and standard deviation of 25. Photo 12 is the tessellation given by the above procedure containing blocks sizes from 4 by 4 to 32 by 32 square. Photo 12 shows each block filled in with the representative value from the quad tree. Notice how boundary areas are represented at lower levels by smaller blocks. Photos 14 and 15 are the results when applied to the natural *girl* image. These results seem encouraging and seem to be the natural counterpart to the orientation based tessellation as used by Calway [25] and Todd [24].

## 4. Conclusions and Proposals for future work

This report has considered the two main approaches to image segmentation: boundary based and region based methods. Multiresolution descriptions have been employed in both, although none of the better schemes have sought to combine the two complementary approaches. An investigation into a set of rotation invariant operators was presented. It was shown how structural information could be extracted from an oriented boundary estimate. Such information could be incorporated into a region based segmentation method to improve its performance. The initial steps of a new quad-tree segmentation method, which is able to represent the regions of an image at an optimal scale, have been developed.

### A combined approach?

The idea of using boundary information with a region based segmentation scheme is not a new one (e.g. the work of Hanson and Riseman [30]). Their method merged an edge detection and relaxation scheme with a region formation method. The integration was an intersection of boundary and region results by using the spatial extent of the edge information. They conceded that there would be problems in areas of texture, however.

A more recent attempt by Hong and Shneier [31] determined a 'surroundedness' measure for each pixel from an oriented edge map. In the first instance any complete boundaries are used to identify separate objects then the surroundedness measure is used to determine starting points of a combined region-growing and splitting algorithm.

The performance of the region based method of Spann and Wilson could be improved by incorporating information about boundary structure. Spann has described the problems of boundary erosion within the quad-tree structure where the anisotropic smoothing boundary pixels leads to misclassification, see figure 4.1. Boundary information such as orientation or the presence of a corner structure, could be used to filter anisotropically thus preserving the promontories. It is hoped to continue work on the new segmentation method described in this report to include some form of boundary information. The rest of the segmentation may follow the lines of the Spann and Wilson method with some form of successive boundary refinement procedure.

**References**


1. Wilson, R. G. and Spann, M., *Image Segmentation and Uncertainty,* Wiley & Sons, New York, 1987.

2. Lenz, R., *Reconstruction, Processing and Display of 3D-Images,* pp. 159-190, Phd Thesis, Linköping University, Sweden, 1986.

3. Danielsson, P. E. and Ye, Q. Z., "A New Procedure for Line Enhancement Applied to Fingerprints," in *Contributions to the Development of Machine Vision Algorithms*, Phd Thesis, Linköping University, Sweden, 1989.

4. Knutsson, H., *Filtering and Reconstuction in Image Processing,* Phd. Thesis, Linköping University, Sweden, 1982.

5. Hubel, D. H. and Wiesel, T. N., "Brain Mechanisms of Vision," *Sci. American*, pp. 130-144, September 1979.

6. Marr, D., *Vision,* Freeman Press, San Francisco, CA, 1982.

7. Marr, D. and Hildreth, E., "Theory of Edge Detection," *Proc. Roy. Soc. London*, vol. B207, pp. 187-217, 1980.

8. Gonzalez, R. C. and Wintz, P., *Digital Image Processing,* Addison Wesley, Reading, 1987.

9. Papoulis, A., *Signal Analysis,* McGraw-Hill Book Co., New York, 1986.

10. Rosenfeld, A. and Thurston, M., "Edge and Curve Detection for Visual Scene Analysis," *IEEE Trans. Comp. C-20*, vol. 5, pp. 562-569, 1971.

11. Canny, J., "A Computational Approach to Edge Detection," *IEEE Trans. PAMI-8*, pp. 679-698, 1986.

12. Knutsson, H. E., Wilson, R. G., and Granlund, G. H., "Anisotropic Nonstationary Image Estimation and its Applications," *IEEE Trans. COM-31*, pp. 338-397, 1983.

13. Ballard, D. H. and Brown, C. M., *Computer Vision,* Prentice Hall, New Jersey, 1982.

14. Ballard, D. H., "Generalizing the Hough Transform to detect arbitary shapes," *Pattern Recognition 13*, vol. 2, pp. 111-122, 1981.

15. Spann, M. and Wilson, R. G., "A Quad-Tree Approach to Image Segmentaion which combines Statistical and Spatial Information," *IEEE PAMI-18*, pp. 257-269, 1985.

16. Chen, P. C. and Pavlidis, T., "Image Segmentaion as an Estimation Problem," in *Image Modelling*, ed. A. Rosenfeld, pp. 9-28, Academic Pr., New York, 1981.

17. Burt, P. J., Hong, T. H., and Rosenfeld, A., "Segmentation and Estimation of Image Region Properties Through Cooperative Hierarchical Computation," *IEEE Trans. Sys. Man Cyber SMC-11*, pp. 802-809, 1981.

18. Hong, T. H. and Rosenfeld, A., *Compact Region Extraction Using Weighted Pixel Linking in a Pyramid,* pp. 222-229, IEEE Trans. PAMI-6 , 1984.

19. Horne, C. and Spann, M., "Region extraction using a dynamic thresholding pyramid," *Proc. Visual Comm. & Im. Proc, SPIE*, vol. 1001, pp. 607-615, 1988.

20. Wilson, Roland G. and Spann, M., "Finite Prolate Spheroidal Sequences and their Applications I, II.," *IEEE Trans. PAMI-10*, 1988.

21. Attneave, F., "Some informational aspects of visual perceception," *Psychological Review, 61*, pp. 183-193, 1954.

22. Asada, H. and Brady, M., "The Curvature Primal Sketch," *IEEE Trans. PAMI-8*, pp. 2-14, 1986.

23. Bårman, H., *Curvature Estimation and Description,* Thesis No. 167, Linköping University, Sweden, 1989.

24. Todd, M., *Image Data Compression Based on a Multiresolution Signal Model,* Phd. Thesis, University of Warwick, Coventry, UK, 1989.

25. Calway, A., *The Multiresolution Fourier Transform: A General Purpose Tool for Image Analysis,* Phd Thesis, University of Warwick, Conventry, UK, 1989.

26. Clippingdale, S., *Multiresolution Image Modelling and Estimation,* Phd Thesis, University of Warwick, Coventry, UK, 1988.

27. Clippingdale, S. C. and Wilson, R. G., ''Quad-Tree Image Estimation: A New Image Model and its Application to Minimum Mean Square Error Image Restoration,'' *Proc. 5th Scand. Conf. on Image Analysis*, Stockholm, 1987.

28. Bigün, J., *Local Symmetry Features in Image Processing,* Phd. Thesis, Linköping University, Sweden, 1988.

29. Abramowitz, M. and Stegun, I. A., *Handbook of Mathematical Functions,* Dover Publications Inc., New York, 1972.

30. Hanson, A. R. and Riseman, E. M., ''Segmentation of Natural Scenes,'' in *Computer Vison Systems*, Academic Press, 1978.

31. Hong, T. H. and Shneier, M., *Extracting Compact Objects Using Linked Pyramids,* pp. 229-237, IEEE Trans. PAMI-6 , 1984.

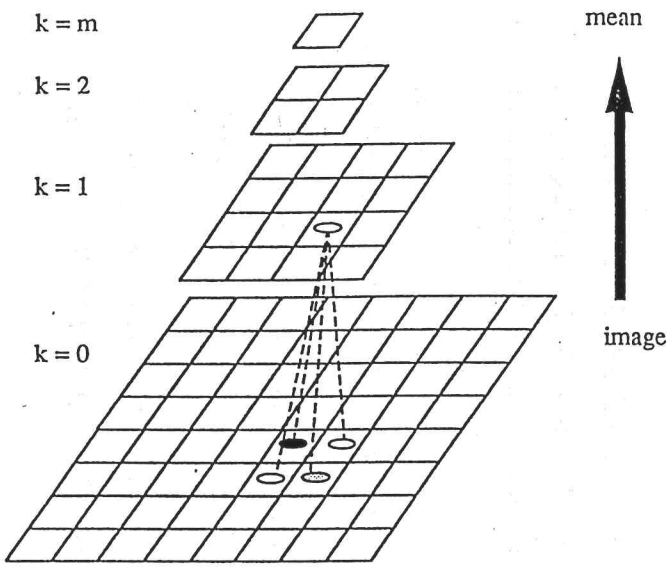


Figure 1.1 A quad-tree of an image size N = 8 , with m = 3 levels

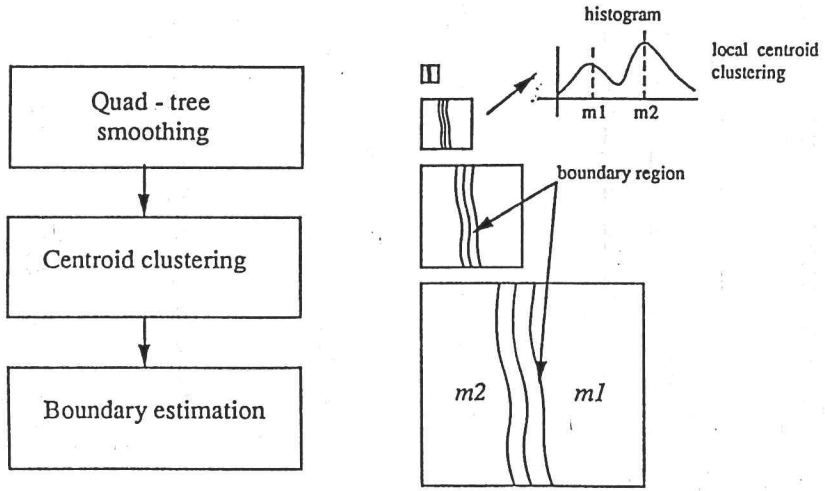


Figure 1.2 Quad-tree segmentation algorithm of Spann & Wilson. m1 and m2 are the classes found by the clustering

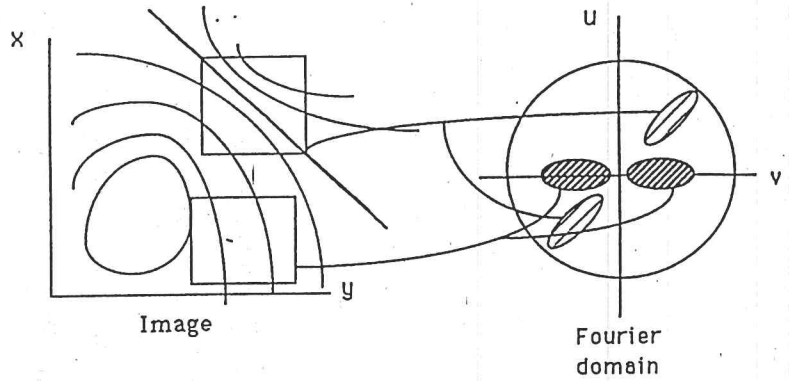


Figure 2.1 Two neighbourhoods and their corresponding neighbourhoods in the Fourier domain

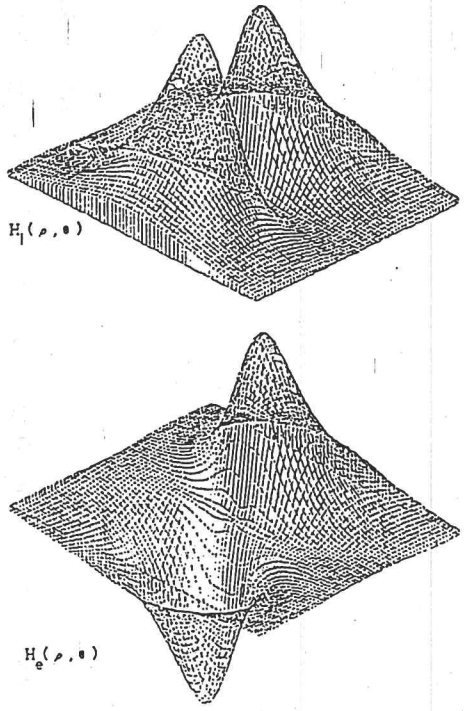

Figure 2.2 One pair of quadrature line and edge filters for orientation estimation.

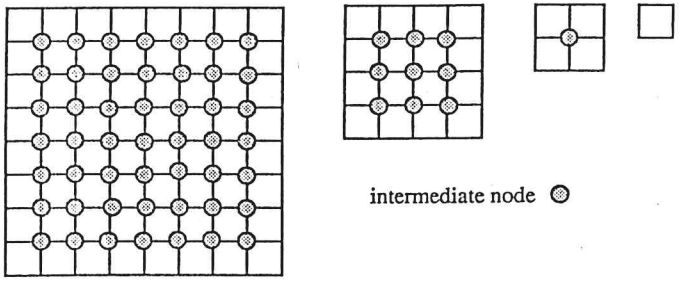


Figure 2.3 Quad-tree showing intermediate nodes

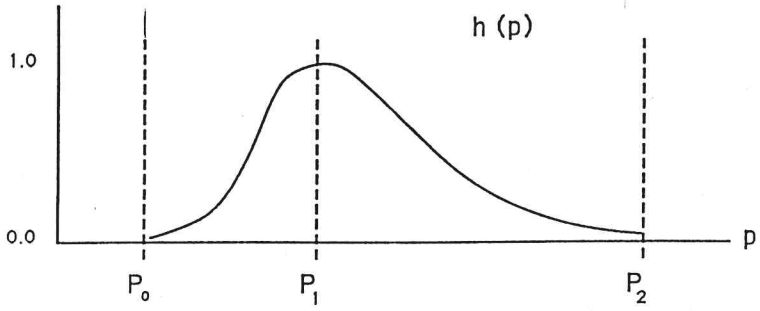


Figure 2.4 Envelope function h(p)

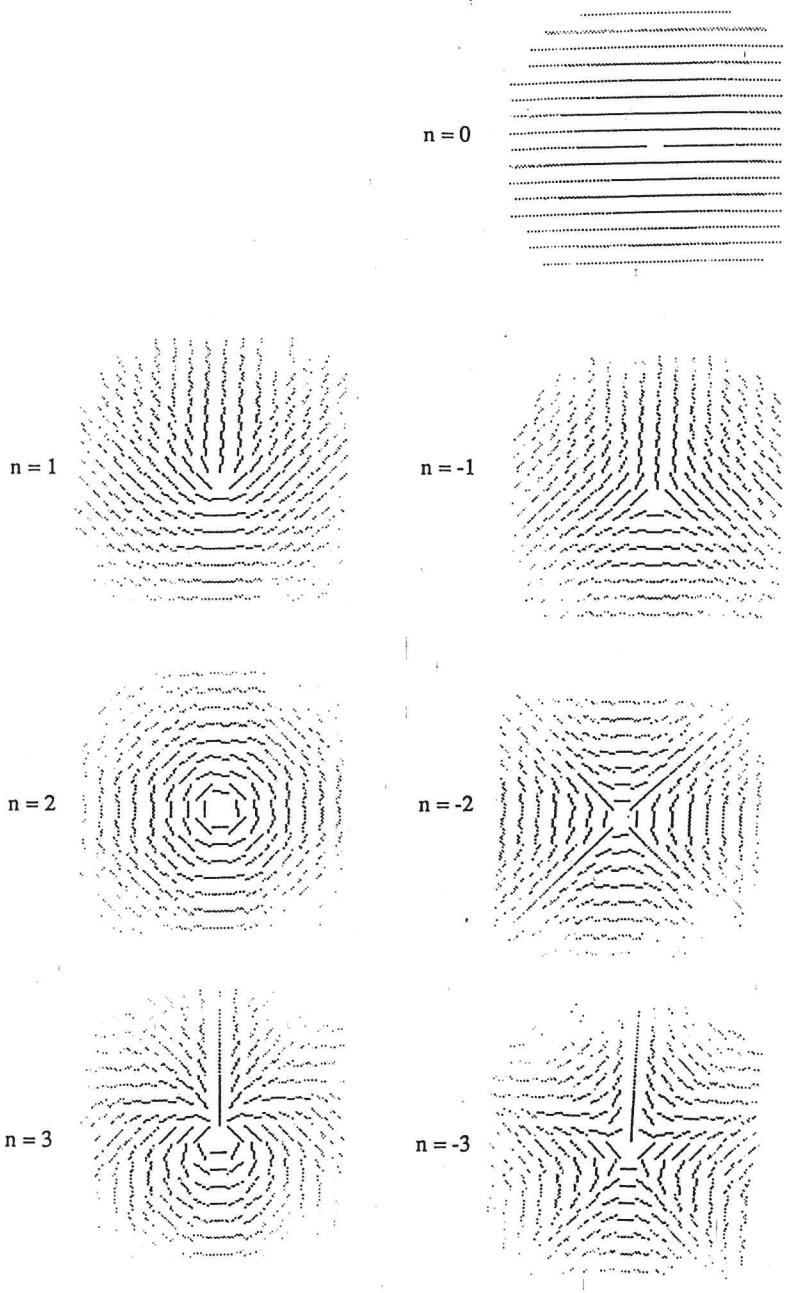


Figure 2.5 Rotation-invariant operators shown as vectors

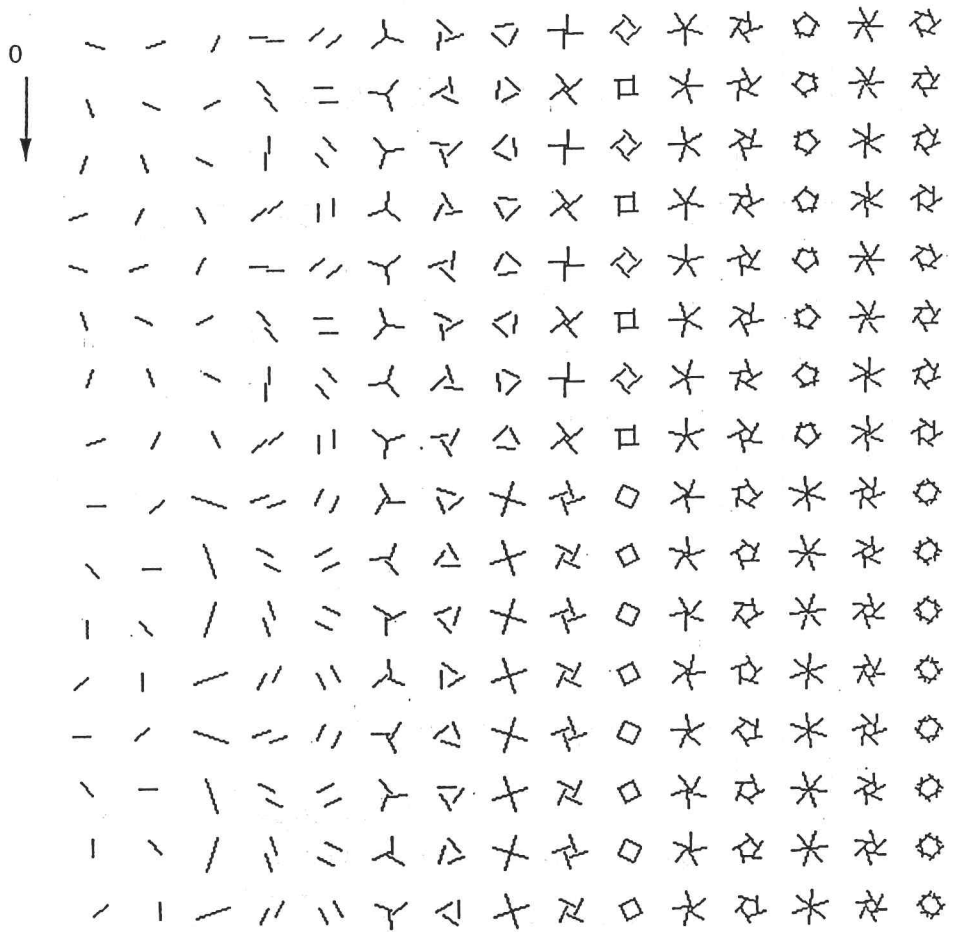


Figure 2.6 Perodic test patterns (numberd from 0, top to bottom left to right)

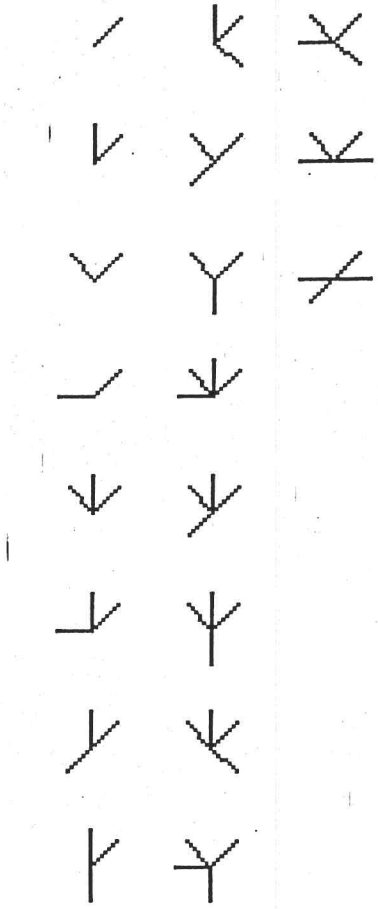

Figure 2.7 Primitives for pattern matching

Figure 2.8 Result of pattern matching

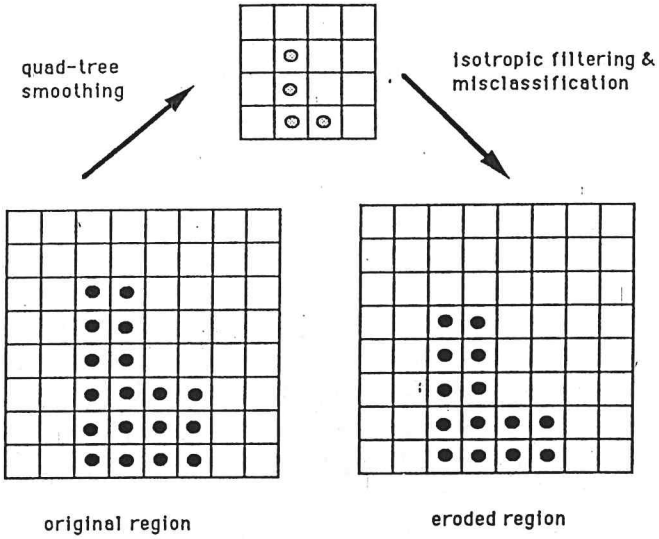


Figure 4.1 Boundary erosion

**Photo 1** Original *girl* image (256x256)

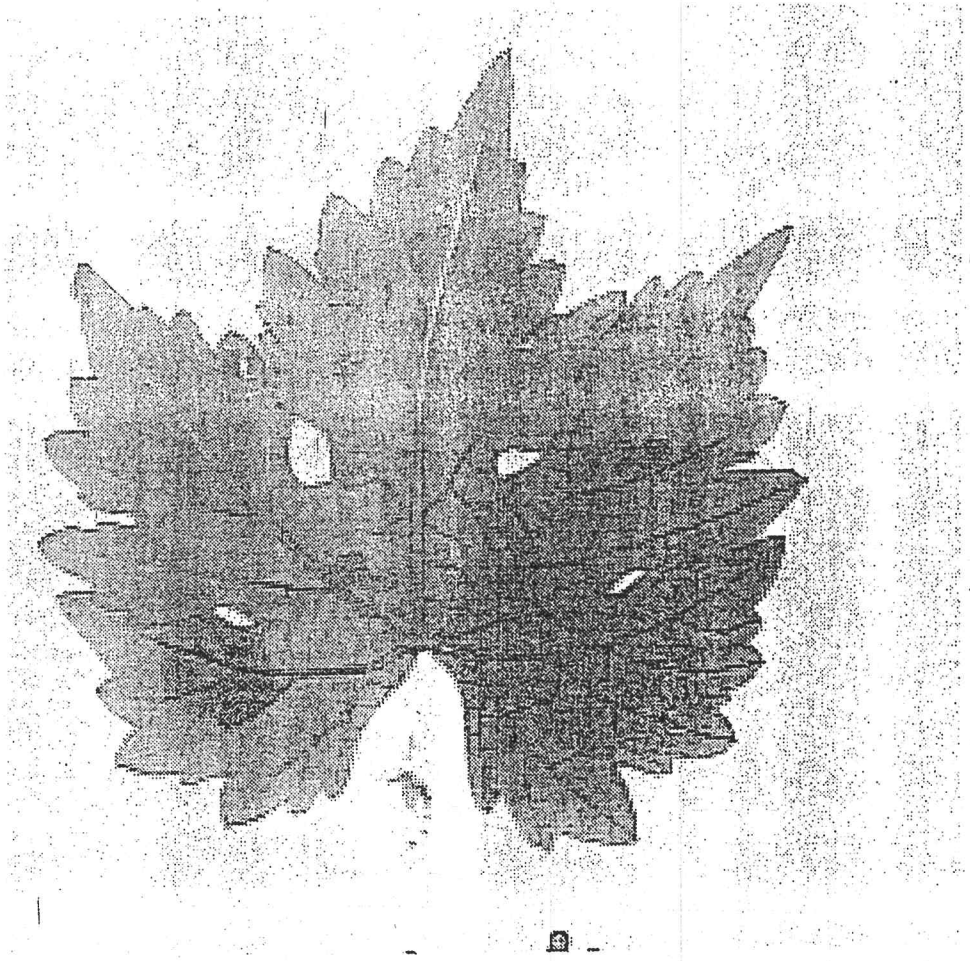

**Photo 2** Original *leaf* image (256x256)

**Photo 3** Magnitude of orientation estimate of *girl*

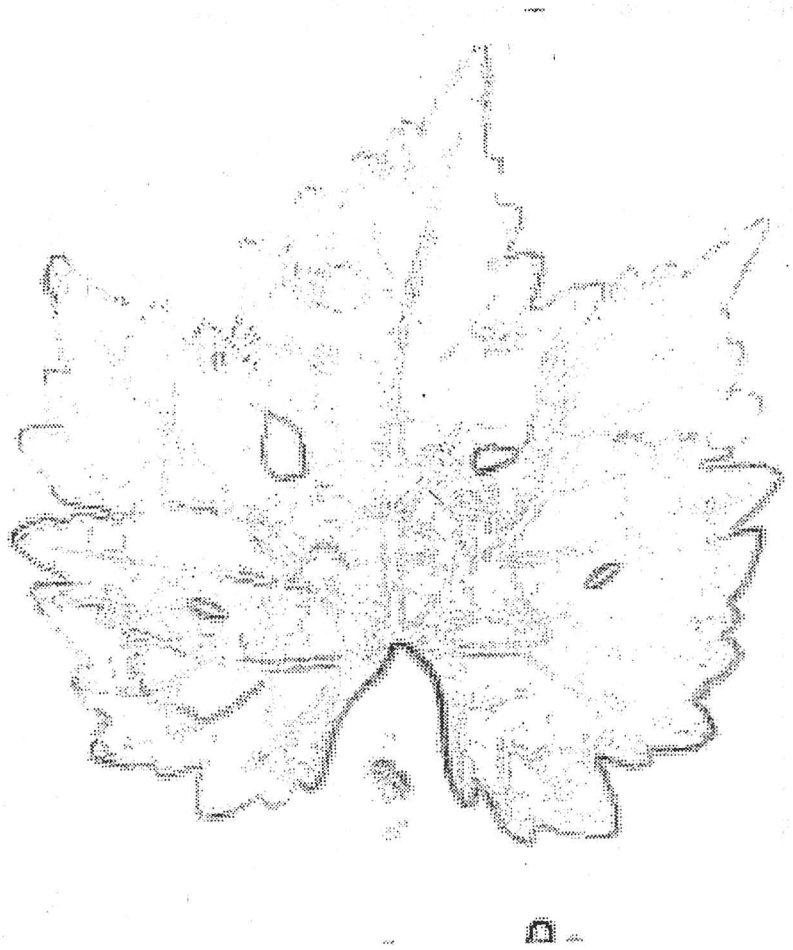

**Photo 4** Magnitude of fast orientation estimate of *leaf*

**Photo 5** Quad-tree normalised, fast orientation estimate of *leaf*

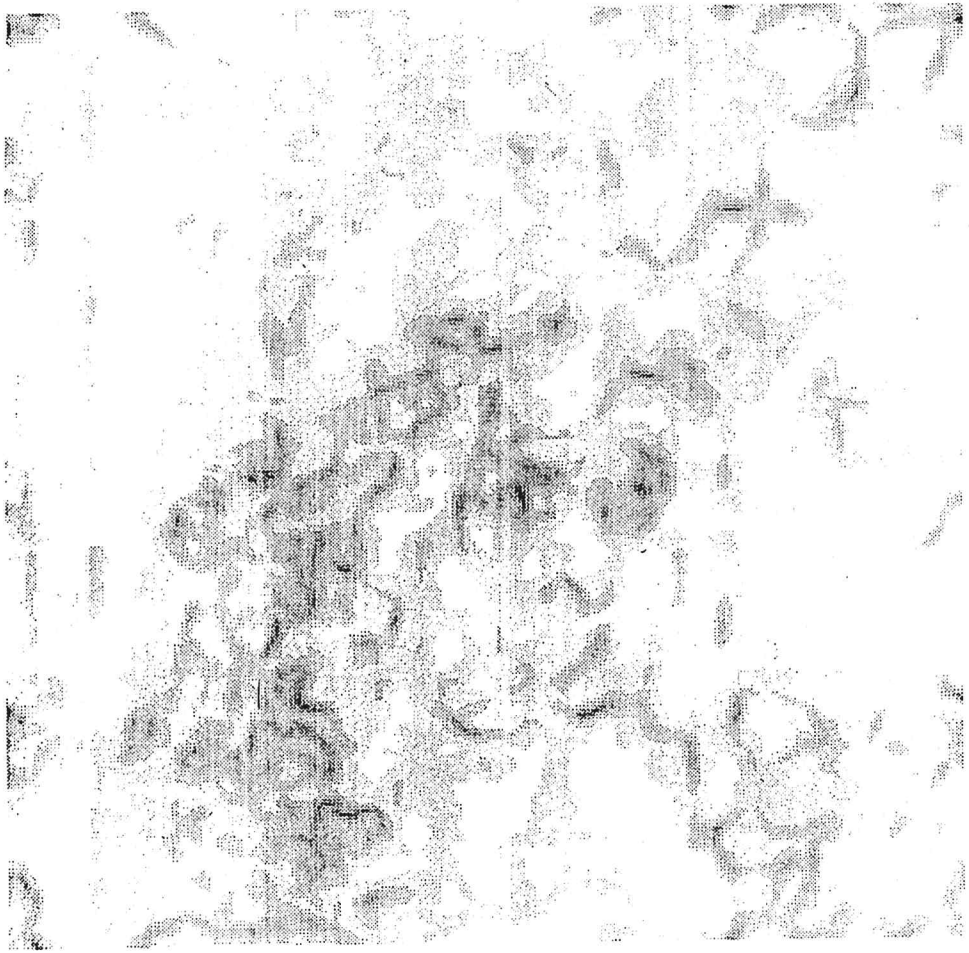

**Photo 6** Inconsistency of orientation of *girl*

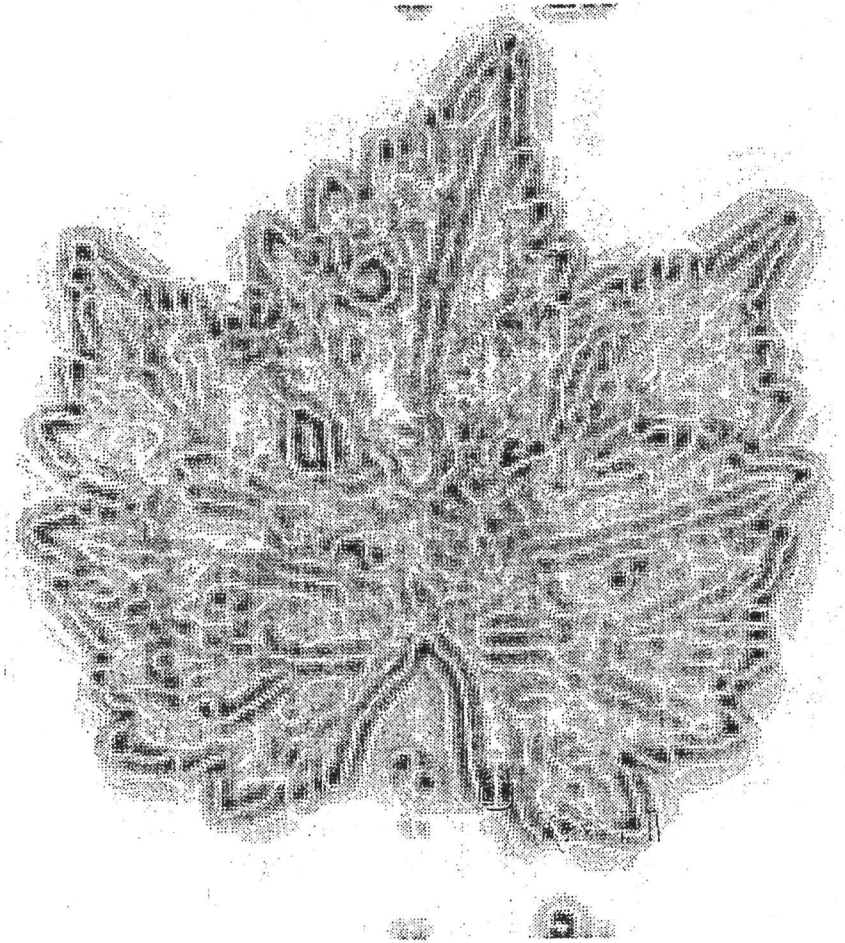

**Photo 7** Rotational operator n = 1 on *leaf*

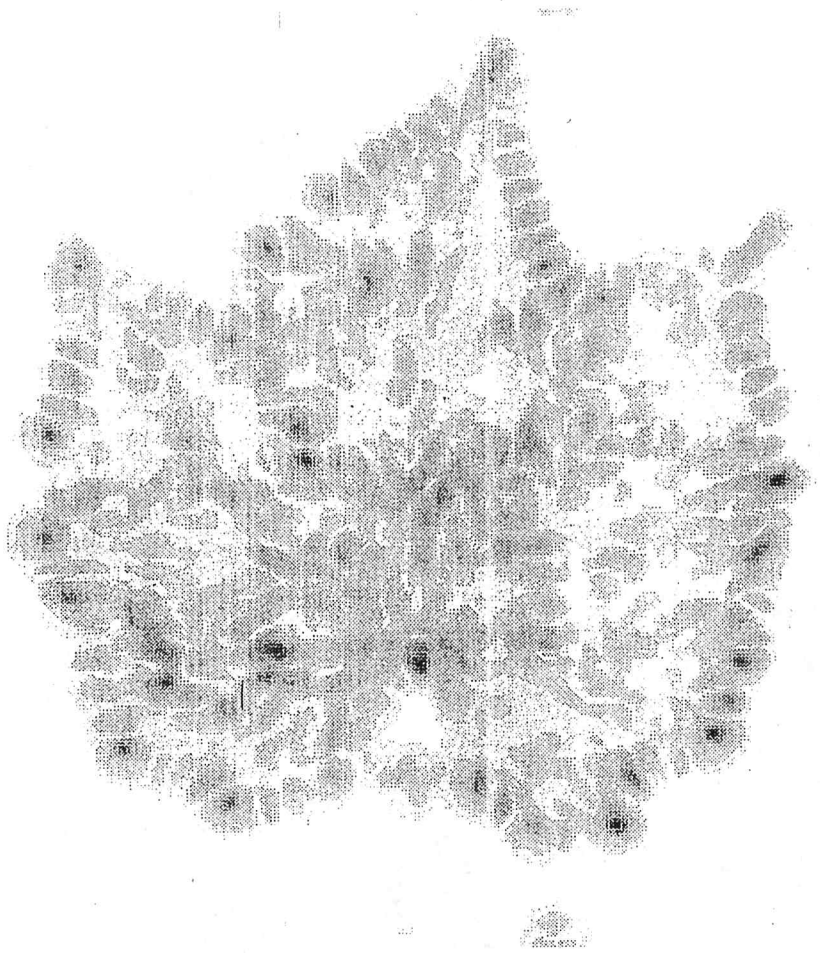

**Photo 8** Auto-normalised rotational operator n = 1 on *leaf*

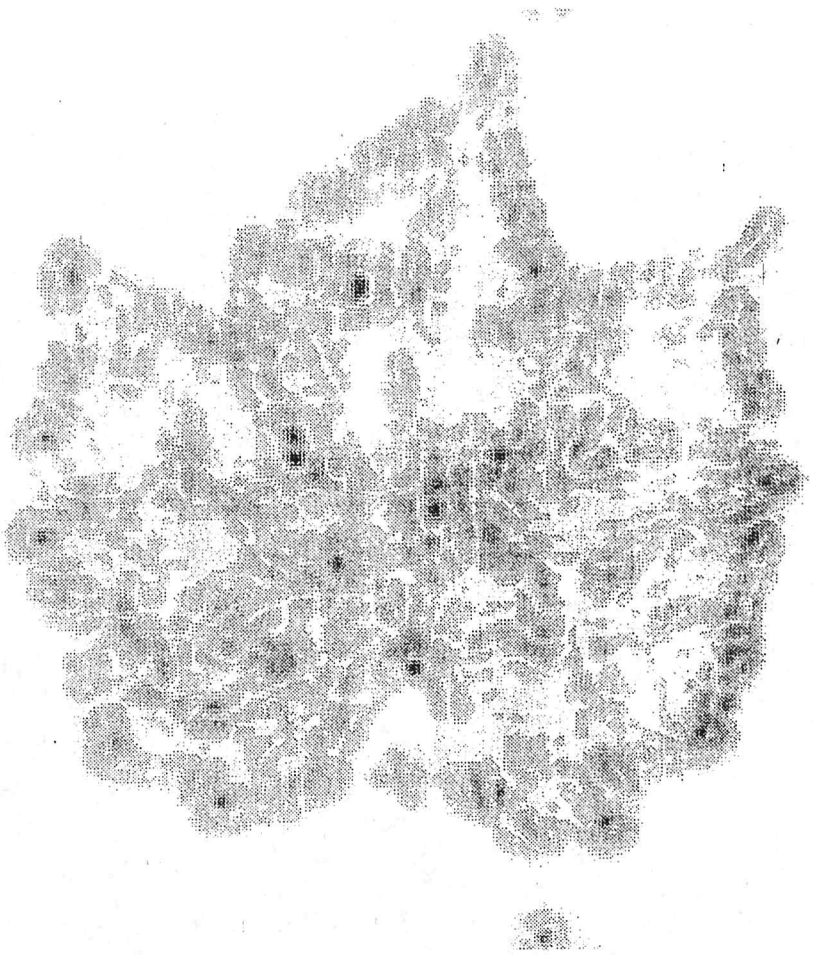

**Photo 9** Rotation operator n = 2 on *leaf*

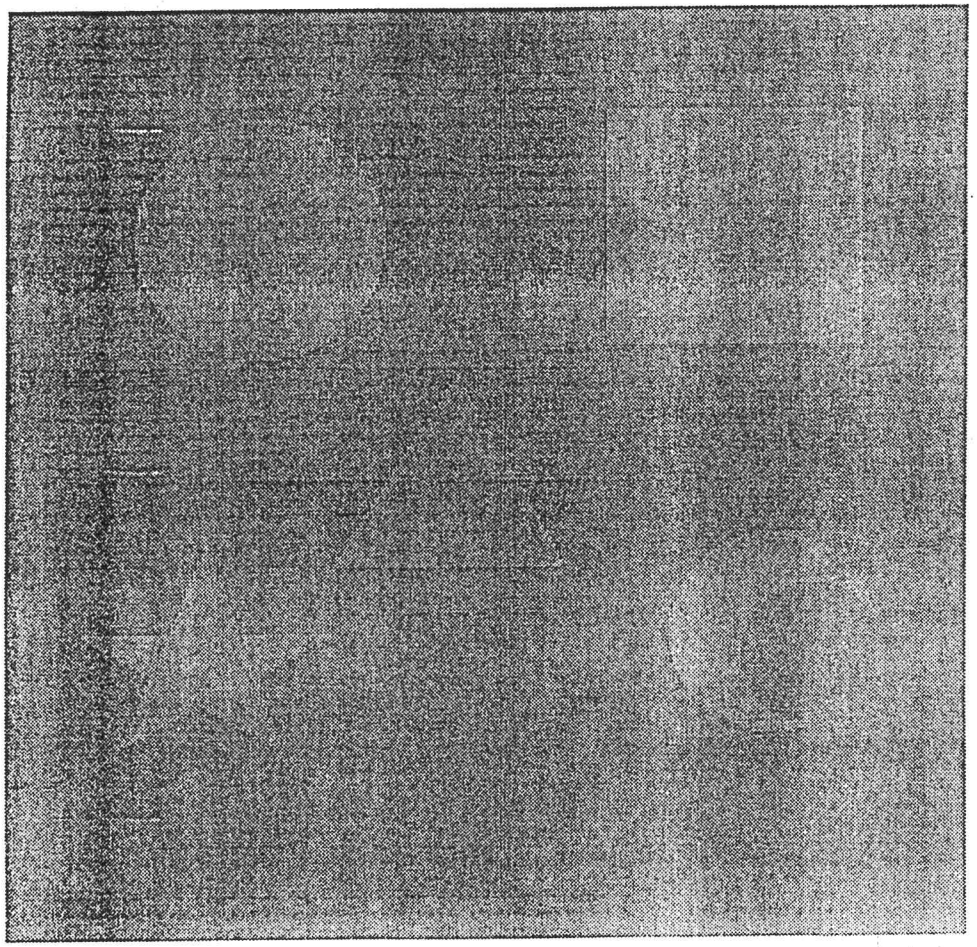

**Photo 10** Original *shapes* image

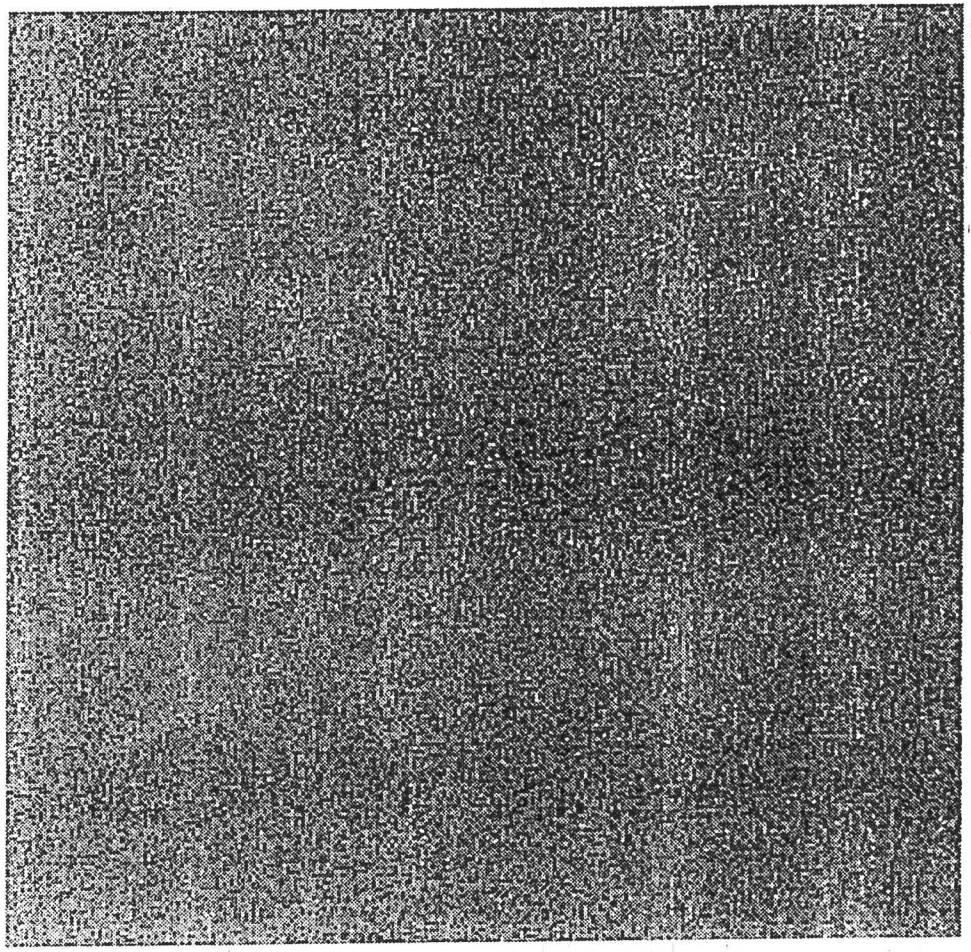

**Photo 11** Noisy *shapes* image, Gaussian white noise sd = 25

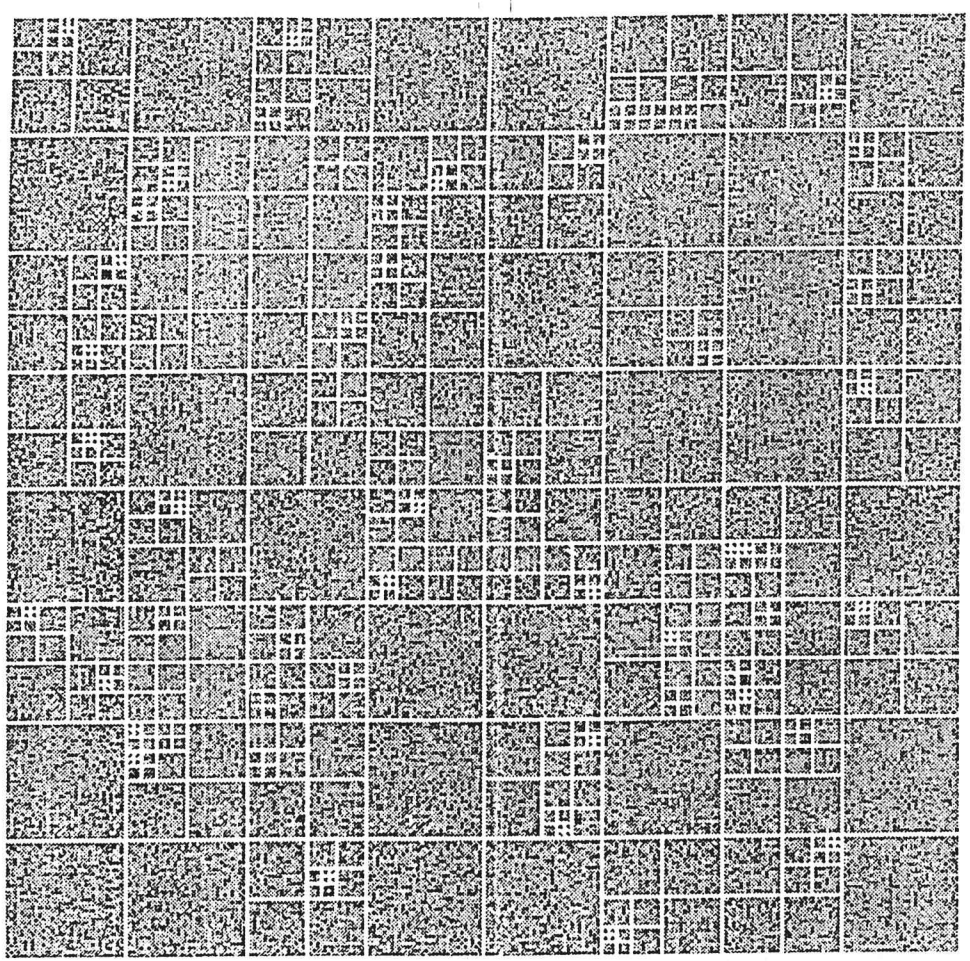

**Photo 12** Quad-tree tessellation of *shapes* image

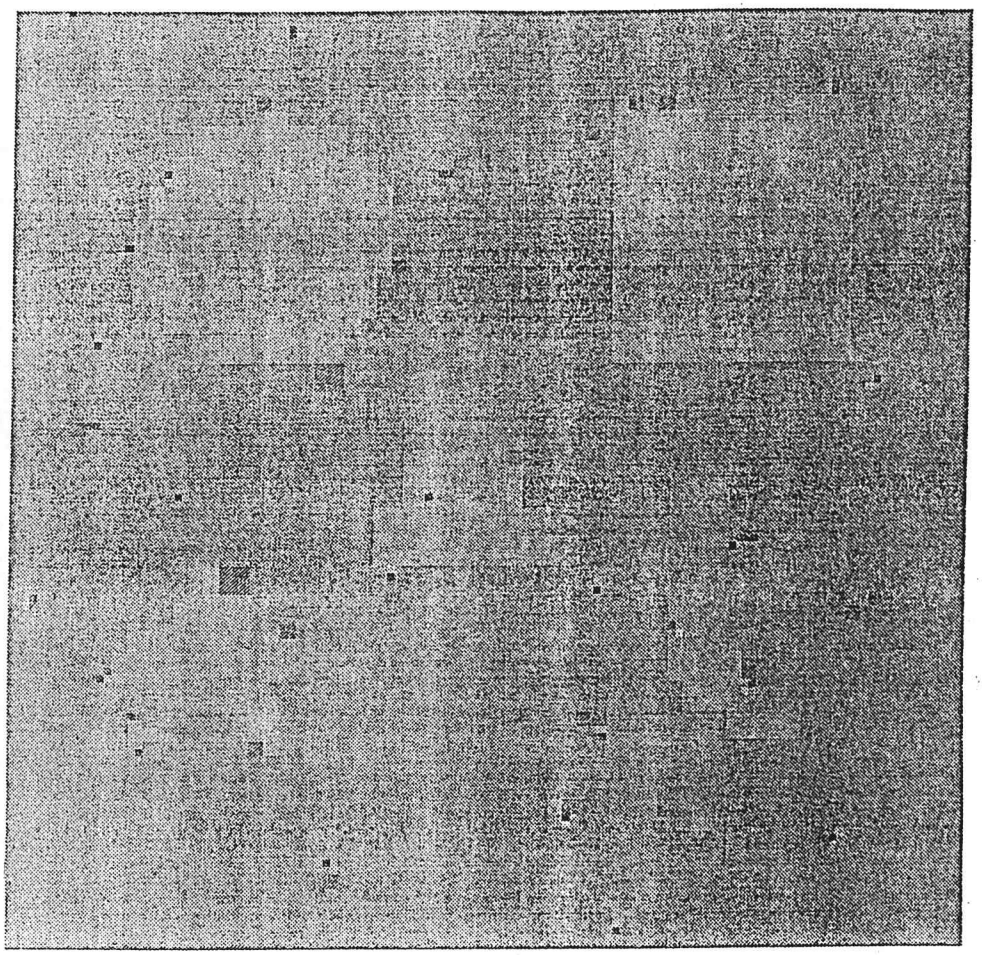

**Photo 13** Quad-tree mean values of tessellation of *shapes* image

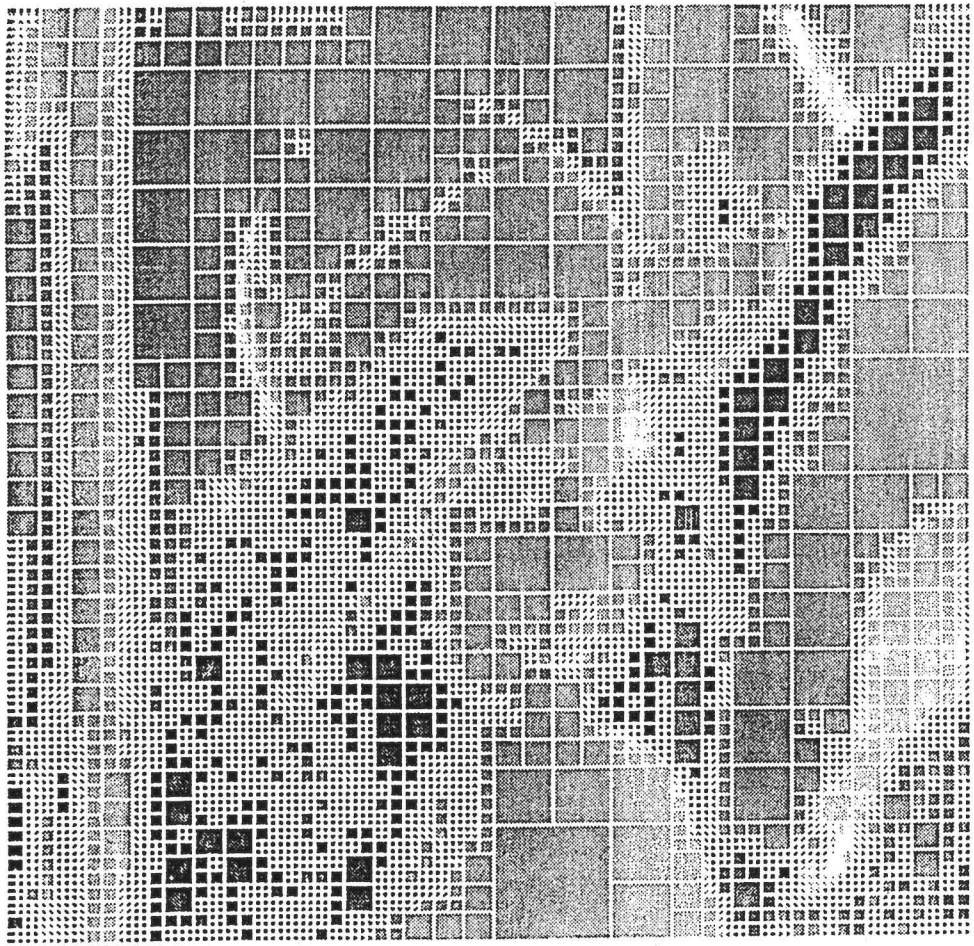

**Photo 14** Quad-tree tessellation of *girl* image

**Photo 15** Quad-tree mean values of tessellation on *girl* image